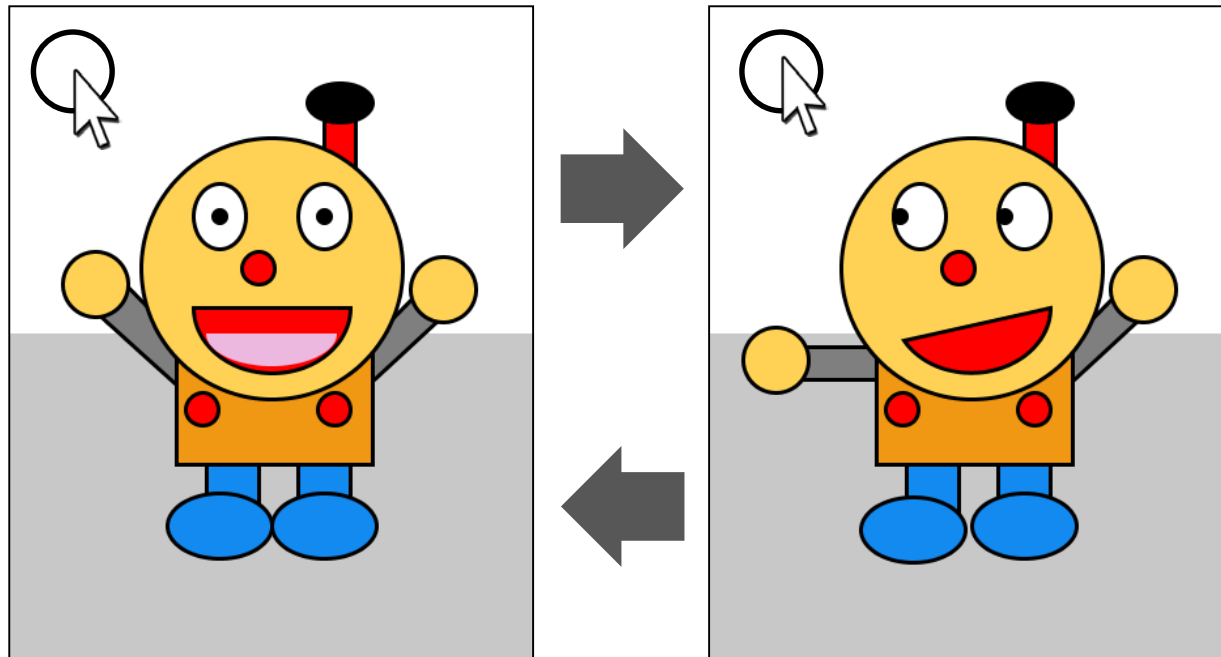


プログラミング演習I (第4回) 課題

• 基本課題① スケッチ名：**charapara**

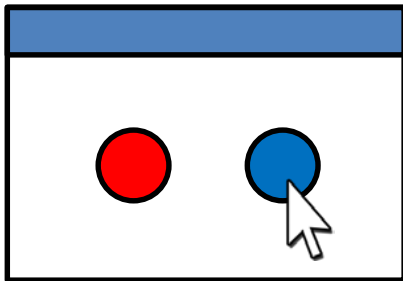
- キャラクタを表示し、画面左上に用意した直径50以下の丸ボタンをクリックすることで絵がパラパラと切り変わるプログラムを作れ。丸ボタンの外をクリックした場合は反応しなくなるようにせよ。
- 前々回の chara1 と chara2 の drawCharacter1 と drawCharacter2をうまく再利用しよう。



プログラミング演習I (第4回) 課題

• 基本課題② スケッチ名： gachaoutput

- ウィンドウに赤色と青色の2つのボタンを表示し，左のボタン（赤色）を押すと5枚のカードの中から1枚がランダムに選択されるプログラムを作成せよ．また，ここでどのカードが選ばれたか出力せよ．さらに，右（青色）のボタンを押すと，その枚数の分布を表示するようにせよ
 - 値をランダムに生成するには，random を使えば良い
 - `random(0, 3);` で，0以上3より小さい実数値がランダムに生成される



```
収集状況 (0,3,1,4,1)
Card3 をゲット
Card4 をゲット
収集状況 (0,3,2,3,1)
Card1 をゲット
収集状況 (1,3,2,3,1)
```



今日のテクニック

ランダムな値を発生したいときは random を利用しよう！

0以上で10より小さい値をランダムに発生する方法は

```
float num = random( 10 );
```

0以上で5より小さい整数値 (0, 1, 2, 3, 4) を同一の確率でランダムに発生する方法は...

```
int num = (int)random( 5 );
```

(0.000... ~ 4.9999... の値が生成されそれを整数化)

10以上20以下の整数値をランダムに発生する方法は

```
int num = (int)random( 10, 21 );
```

プログラミング演習I (第4回) 課題

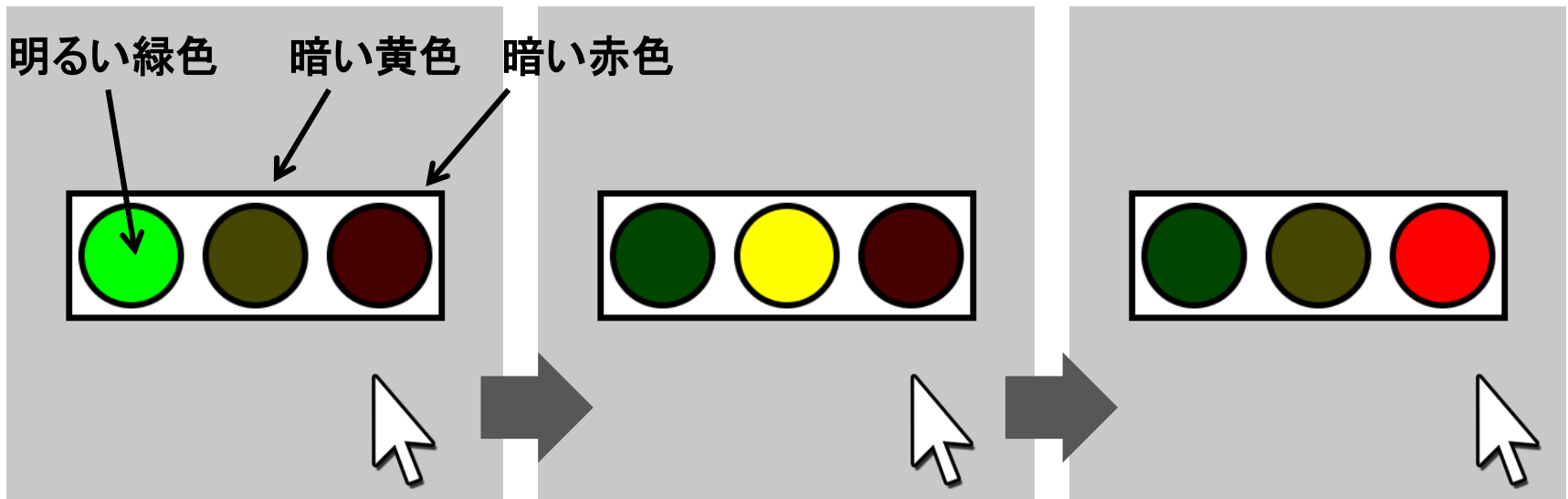
- ヒント

- それぞれの値を管理する変数を用意しよう
- card1, card2, card3, card4, card5 など
- ランダムに整数値を発生させ, その数に応じて card1 ~ card5 の変数の値を増やす
- card1 ~ card5 の値を表示する
- (配列でやってもOK)

プログラミング演習I (第4回) 課題

● 基本課題③ スケッチ名：**signal**

- 信号機のプログラムを作ってください。
- マウスクリックによって光る丸が青→黄→赤→青→... とシフトしていくようにしてください。
- クリックする場所は画面内のどこでも良いです。



画面をクリックしたら色が変わる！

赤の次はまた青に戻るように！

プログラミング演習I (第4回) 課題

発展課題① スケッチ名：launch

- 画面左下からマウスカーソルがある方向にボールを発射する砲台のプログラムを作ってください。
- クリックされたタイミングで投射されるようにすること。また、初速はクリック時の砲台からカーソルまでの距離にすること。さらに、画面の右端まで行くと左方向に跳ね返るようにせよ。跳ね返りの場合はそのまま左右方向を反転させるだけでよい。draw1回あたり0.1秒ずつ進むものとする

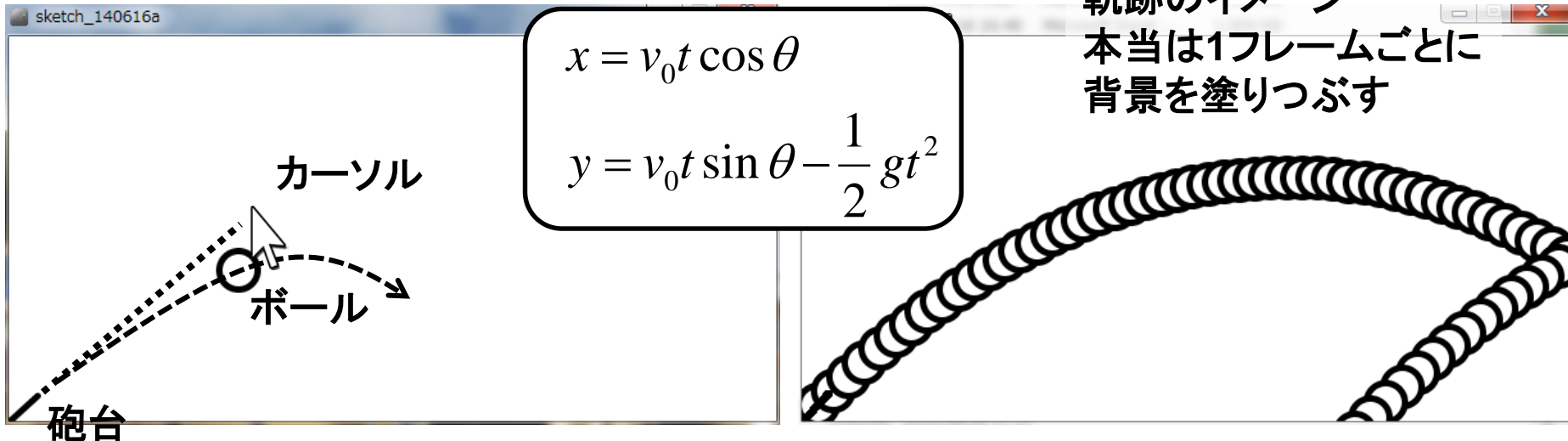
斜方投射の式

$$x = v_0 t \cos \theta$$

$$y = v_0 t \sin \theta - \frac{1}{2} g t^2$$

軌跡のイメージ

本当は1フレームごとに背景を塗りつぶす



ヒント

- この世界を実現する上で必要な変数（変化するものを取りあえず片っ端から）をまず列挙！
 - マウスの座標
 - 砲台の角度
 - ボールの投射角度
 - ボールの初速
 - ボールの座標
 - 時間経過を管理する変数
 - ほかにある？
- その変数によって世界を描画してみる！
- 変数に、計算などによって値を入れてみる
- 跳ね返るということはどういう計算があればよい？

プログラミング演習I (第4回) 課題

• 発展課題② スケッチ名：charajump

- キャラクタと地面となる面を表示し、画面左上に用意した直径50以下の丸ボタンをクリックすることでキャラクタがジャンプするようなプログラムを作成せよ。
- ジャンプについては初速をわかる程度に設定しつつ鉛直上方投射するものとし、地面にくると止まるようにせよ

